Comparing Search Strategies for Finding Global Optima on Energy Landscapes

K. W. FOREMAN, A. T. PHILLIPS, J. B. ROSEN, K. A. DILL

Department of Pharmaceutical Chemistry, University of California at San Francisco, San Francisco, California 94118

Received 22 December 1998; accepted 28 May 1999

ABSTRACT: We provide some tests of the convex global underestimator (CGU) algorithm, which aims to find global minima on funnel-shaped energy landscapes. We use two different potential functions—the reduced Lennard–Jones cluster potential, and the modified Sun protein folding potential, to compare the CGU algorithm with the simplest versions of the traditional trajectory-based search methods, simulated annealing (SA), and Monte Carlo (MC). For both potentials, the CGU reaches energies lower on the landscapes than both SA and MC, even when SA and MC are given the same number of starting points as in a full CGU run or when all methods are given the same amount of computer time. The CGU consistently finds the global minima of the Lennard–Jones potential for all cases with up to at least $n = 30$ degrees of freedom. Finding the global or near-global minimum in the CGU method requires polynomial time [scaling between $O(n^3)$ and $O(n^4)$], on average.
 J Comput Chem 20: 1527–1532, 1999

Keywords: global optimization; global underestimator; conformational searching landscapes; Lennard–Jones clusters

Introduction

Many problems in computational biology, such as the folding of proteins or of RNA molecules and the docking of ligands, require search methods that find the global optima of an energy function. Practical application of these search methods requires them to be both rapid and consistently accurate. Traditional trajectory-based search methods—molecular dynamics, Monte Carlo (MC), and simulated annealing (SA)[1,2]—are fast but often inconsistent in their prediction of global minima because of kinetic traps. Recent developments in global search methods include underestimator methods,[3–8] such as the Convex Global Underestimator (CGU). Rather than searching via trajectories over energy landscapes, global underestimator methods search under the landscapes and are, therefore, not subject to kinetic

Correspondence to: K. W. Foreman; e-mail kw@zimm.ucsf.edu
Contract/grant sponsor: Sloan Foundation, Lawrence Berkeley Lab, and San Diego SuperComputer Center
Contract/grant sponsor: NSF; contract/grant number: DBI-9808321

 CCC 0192-8651 / 99 / 141527-06

trapping. Underestimator methods are based on the assumption that energy landscapes are funnel-like. This assumption is likely to be widely obeyed throughout computational biology[9] and in other physical problems that involve unique stable states.

In this article, we contrast the CGU with three traditional search methods using two different model energy functions. Our aim is to illustrate differences of underestimators compared to trajectory methods for funnel-like landscapes. We compare the CGU to random sampling with minimization, to SA and to MC. Each search method explores landscapes generated from an atomic Lennard–Jones cluster function, and from a simplified continuum model protein folding function. Our key finding here is that the CGU underestimator method is more efficient at reaching more deeply toward the global minima on funnel-like landscapes than the simple implementations of the trajectory-based search strategies that we test here. The efficiency grows with the size of the problem.

Our goal here is not a speed comparison. Although the CGU is fast, any speed comparison we could make to some trajectory-based strategy is subject to criticisms about alternative implementations. In practice, most search methods are not perfectly general. They require some optimization of cooling schedules, crossover rules, acceptance ratios, move sets, etc. The standard procedure is often to adjust the search method by seeking optimal parameters for the given problem. While some problem-specific adjustments could also be made to the CGU, we make none here, nor to any of the other methods. Instead, we choose, as best we can, the simplest, most canonical implementation of each method. This choice of implementations is intended more to illuminate the performance and principles of the CGU than as a critical comparison against existing strategies or implementations.

Summary of Search Methods

Although well known, we briefly review the MC and SA search methods here. Both methods begin at an arbitrary point on the energy landscape. The MC method then takes a random step from the starting point. The new point becomes the next starting point with probability $P(E_{\text{new}}, E_{\text{old}})$,

$$P(E_{\text{new}}, E_{\text{old}}) = \min(1, \exp([E_{\text{old}} - E_{\text{new}}]/kT)), \quad (1)$$

where $\min(x, y)$ is the smaller of x and y, k is Boltzmann's constant, T is the temperature in Kelvin, and E_{new} and E_{old} are the energies evaluated at the new and previous points, respectively. If the new point is rejected, the previous point remains the starting point. The random step and selection of the next starting point constitutes a MC step. We use 10^7 MC steps with a temperature-dependent maximum step size s,

$$s = \Delta L\sqrt{T}, \quad (2)$$

where ΔL is the minimum range in any dimension, divided by 1000. This choice of ΔL produces a maximum step size equal to the smallest range of all dimensions when $T = 10^6$ K. The lowest energy found typically does not change much after 50,000 MC steps. The actual step size in each dimension is determined by multiplying s by a random number between -1 and 1. Calculations are performed at $T = 300$ and 1000 K, and selection of points is constrained to remain within a bounding box appropriate for the given test problem. The lowest energy found from all the MC steps is returned as the putative global minimum. This approach gets trapped in local minima if barriers are larger than kT.

SA is similar to MC, but attempts to overcome local trapping by starting at a high temperature and following multiple trajectories simultaneously. The starting points for the trajectories are randomly selected around a user specified point. We choose points randomly within a bounding box appropriate for the given test problem. An annealing or ''cooling'' schedule is designed to focus the search in regions of low energy. In principle, this SA routine[2] (also see http://www.ingber.com for more details) can always find the global minimum, but the time required typically grows exponentially with the size of the search space. We use the default exponential cooling schedule, and do not allow trajectories to leave the bounding box.

The CGU method operates very differently than either SA or MC methods. Each CGU step has three stages. In the first stage of a CGU step, points on the energy surface are chosen randomly and local minima are found. In the second stage of a CGU step, the true energy landscape $F\{\mathbf{x}\}$ is approximated by an underestimating parabola $U\{\mathbf{x}\}$,

$$U(\mathbf{x}) = a_0 + \sum_{i=1}^{n} a_i x_i + d_i x_i^2, \quad (3)$$

where $U\{\mathbf{x}\}$ is the energy of the parabola at point $\mathbf{x}$, n is the number of degrees of freedom, and the a_is and d_is are constants determined within the algorithm as described below. The d_is must all be nonnegative to guarantee convexity. The numerical values for these constants are determined through linear optimization such that the parabola either hits or underestimates by a minimum amount all known minima on $F\{\mathbf{x}\}$ (see Fig. 1). The function we minimize is,

$$\sum_j \left(F\{\mathbf{x}_j\} - U\{\mathbf{x}_j\}\right), \tag{4}$$

subject to the constraint that $F\{\mathbf{x}_j\} - U\{\mathbf{x}_j\} \geq 0$ for all points $\mathbf{x}_j$. This functional form has the advantages that it both requires the least number of points $(2n + 1)$ to generate a convex function, and helps pick subsequent bounding boxes (see below). The default number of points used in the first stage of a CGU step is four times the minimum (i.e., $8n + 4$ points), although using as few as $2n + 1$ points does not appear to degrade the final results significantly. Having found the a_is and d_is, the third stage of a CGU step involves choosing a new search window, centered about the parabolic minimum, for the next CGU step.

The key to the localization of the conformational search by the CGU is the narrowing of the bounding box from one CGU step to the next. A parabolic energy function (such as $U\{\mathbf{x}\}$) has a Gaussian probability distribution for the lowest energy state. The center of the bounding box is chosen to be the minimum of $U\{\mathbf{x}\}$, and the width of the bounding box is chosen to be $\sigma_i \equiv d_i^{-\frac{1}{2}}$, one standard deviation of the Gaussian function for degree of freedom i. If the CGU's current best local minimum lies outside the window, the bounds are widened to include it. Even if the global minimum of $F\{\mathbf{x}\}$ does not lie within the boundaries for the next CGU step, the underestimate in the next CGU step will be strongly influenced by the tight clustering of points. If the landscape is funnel-like, the effect on the parabola shape will be either to push the parabolic minimum towards the global minimum or to increase the width of the next bounding box. CGU steps are taken until either the coordinates for the minimum of $U\{\mathbf{x}\}$ correspond to those for the current best local minimum or the lowest energy minimum has not changed for five steps, at which point the lowest energy minimum is taken to be the putative global minimum. Typically, no more than 10 CGU steps are required. We find that the CGU finds its solutions in polynomial time, on average, but this is not guaranteed.

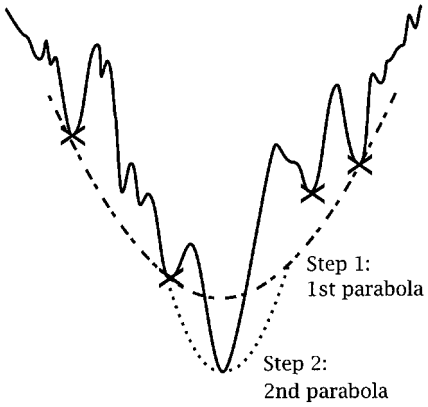

FIGURE 1. Illustration of the CGU method.[8] Starting points are randomly selected and locally minimized (X's) along the energy surface (solid line). A quadratic surface (dashed line) either hits or is lower in energy than any local minima known at that stage. The location of the quadratic minimum becomes the center of another box in which another round of sampling is performed. For funnel-like surfaces, the single minimum on the quadratic surface lies near the global minimum of the energy function. In each subsequent step, a smaller region of space is sampled on average.

Lennard–Jones Clusters

For one test, we use the reduced Lennard–Jones (LJ) potential,

$$E_{\mathrm{LJ}} = \sum_{i=1}^{N} \sum_{j>i} \left(\frac{1}{r_{ij}^{12}} - \frac{2}{r_{ij}^{6}} \right), \tag{5}$$

where r_{ij} is the distance between the centers of atoms i and j, and N is the number of atoms. Equation (5) is useful because it has been widely studied in global optimization problems. Its global minima are well known for small clusters. The search time to find the global minima is expected to grow exponentially due to the increase in number of local minima that crowd the global solution. LJ clusters in a reduced potential have a combinatorial degeneracy in the number of global minima, because any two atoms in the global minimum configuration can be switched without changing the energy calculated from eq. (5). Thus, the energy surface contains a large number of equivalent funnels, and each funnel has many local minima with small root-mean squared deviation from the closest global minimum. The reduced LJ potential

of Eq. (5) thus provides a challenging test of global minimization methods.

We apply one constraint to keep the search manageable. We assume that the atoms lie within a bounding box whose edge length is roughly twice the minimum necessary to bound the best known global solution.[10] This same constraint is applied uniformly to all the search methods.

For each standard method, we randomly assign the initial coordinates of each atom within the bounding box. The first atom is fixed at the origin, the second atom is constrained to lie along the x axis, and the third atom has only its z coordinate fixed at zero. All remaining atoms are free to be placed anywhere within the cube. Calculations are performed for clusters with 2, 3, 4, 5, 6, 9, 14, and 21 atoms. No other assumptions are made.

We compare the three methods using two different bases. First, we compare the results if all methods are run for identical dedicated times on the same computer. Second, we compare the result using an equivalent samples basis: given a number of samples used by the CGU, our second comparison allows SA and MC to use this number of randomly chosen starting points to attempt to find global minima. By either comparison, the CGU method consistently finds states of lower energy than the other two methods. The CGU finds a global minimum for all but the 21 atom cluster. In contrast, the SA and MC methods only find global minima for clusters with up to nine and four atoms, respectively (see Fig. 2). Better performance using the SA and MC algorithms has previously been reported.[11–13] But, again, we note that our aim here is not to develop an optimal strategy for van der Waals clusters, or any other specific application. Rather, our aim is to give an equitable test of unoptimized strategies on general funnel landscape problems. By using only the simplest implementations, we reduce the arbitrary choices we must make. The CGU solutions from multiple trial runs have identical (global minimum) energies but permuted coordinates, implying that the CGU has no bias among the equivalent global minima for van der Waals clusters. The equivalent global minima create a landscape that is not particularly funnel-like. The existence of multiple funnels is reflected in all the d_is being nearly zero, with the consequence that the search bounds do not constrict. This lack of focussing implies that the CGU in this problem is no more effective than random sampling with subsequent local minimization. We expect that performing local minimization after each SA or MC step would have considerably

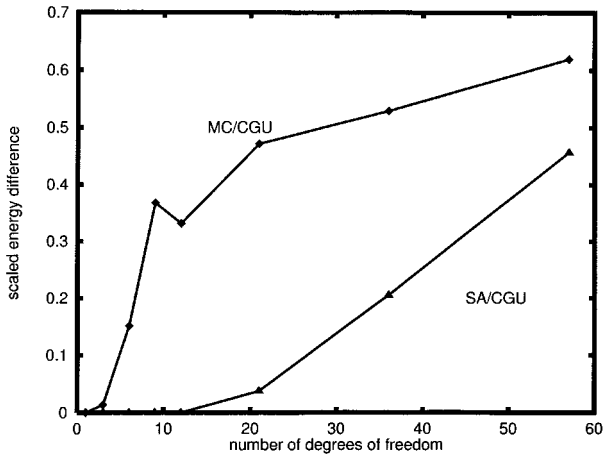

FIGURE 2. Scaled energy difference $(E_X - E_{CGU}) / |E_{CGU}|$ [X = SA(triangles), MC(diamonds)] as a function of the number of degrees of freedom, E_X and E_{CGU} are the average energies found using method X and the CGU method, respectively, on the potential in eq. (5) for clusters of atoms. The CGU method is more effective than either the SA or MC methods in searching for low-energy clusters with four or more atoms.

improved their performance as well. These results suggest that whenever the landscape is not globally funnel-like, the CGU is only as good as the sampling technique used in the first stage of each CGU step.

The CGU method is reasonably fast. The CGU method runs with an average case time scaling as $O(n^3)$ to $O(n^4)$ (See Fig. 3), as expected from previous work.[8] The SA method has a short run time, typically requiring no more than a minute for each starting configuration of the 21 atom cluster. The MC method runs slower than SA, by as much as a factor of 5. Both SA and MC typically require a much larger number of separate runs to find global minima as the dimensionality of the problem increases, thus requiring a larger total run time than the single CGU run. Adjusting the temperature or number of cycles in the MC method did not increase speed or accuracy. Likewise, slowing the annealing schedule in the SA method enormously increases the time required for each run without significantly improving the final predicted energy.

Protein Folding Model of Sun et al.

We have also compared the three search methods on the modified Sun potential[7,14] for protein folding. This model explicitly includes backbone

atoms, but represents side chains as either hydrophobic or hydrophilic spheres whose radius is a function of the amino acid. The energy E of a conformation is a sum of four components, a hydrophobic term E_{hp}, a hydrogen bonding term E_{hb}, an excluded volume term E_{ex} and a penalty $E_{\phi\psi}$ for ϕ/ψ pairs that lie outside permitted regions of the Ramachandran plot,

$$E = E_{hp} + E_{hb} + E_{ex} + E_{\phi\psi}. \quad (6)$$

The hydrogen bonding term E_{hb} is calculated via electrostatics for each backbone N—H group, which is less than 25 angstroms from a given backbone carbonyl oxygen,

$$E_{hb} = 1656.05 \sum_{i=\{C,O\}} \sum_{j=\{N,H\}} \frac{q_i q_j}{d_{ij}}, \quad (7)$$

where $q_C = -q_O = 0.5$, $q_H = -q_N = 0.3$, and d_{ij} is the distance between atoms i and j. The hydrophobic term is a sum over the locations of all pairs of hydrophobic side-chain centroids,

$$E_{hp} = \sum_{i,j,|i-j|>2} \frac{-1}{1 + \exp(.4 d_{ij} - 2.6)}, \quad (8)$$

where i and j range over the hydrophobic residues, and d_{ij} is the distance between the side-chain centroids of residues i and j. The excluded volume term E_{ex} takes the form,

$$E_{ex} = \sum_{i>j} \frac{5}{1 + \exp(10 d_{ij} - 36)}, \quad (9)$$

where i and j range over the labels of both the atoms and centroids in the model and d_{ij} is the distance between elements i and j, while the Ramachandran penalty $E_{\phi\psi}$ is two dimensional with an ellipsoid boundary.

$$E_{\phi\psi} = \frac{0.25}{1 + \sum_i \exp[f_i(\phi, \psi)]}, \quad (10)$$

where $f_i(\phi, \psi)$ is given elsewhere.[7] The potential defined in eqs. (6)–(10) is expected to be funnel-like,[7,8] but with a large number of local minima. The virtue of this potential function is that it is very simple and protein-like. The disadvantages are that it is not an accurate folding potential (none is yet known), and its global optima are not known with certainty.

We performed several runs using the modified Sun potential. Because the MC method is poor both in terms of speed and accuracy, Figure 4 compares only SA with the CGU. We permit the SA method to start from as many points as the CGU method randomly samples, because this comparison gives better results for the SA method on the LJ clusters. Figure 4 illustrates the compari-

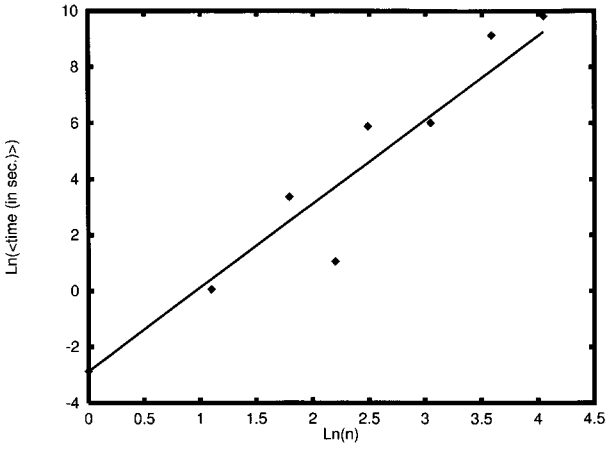

FIGURE 3. Average CGU run time (in seconds) as a function of the number of degrees of freedom (n). The dashed line is a guide and has a slope of three. The CGU method scales as $O(n^3)$ for the simplified Lennard–Jones potential of eq. (5), and has a prefactor of about 0.05 s.

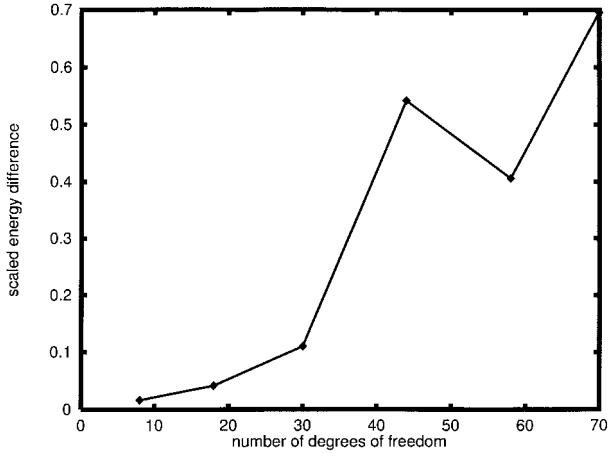

FIGURE 4. Same as Figure 2, except now applied to the Sun et al. protein folding potential. The MC / CGU curve is not included because differences are too large. The CGU method reaches deeper energies, on average, for this potential than the SA method, with the difference growing faster as a function of the number of degrees of freedom for this funnel-like potential than for the degenerate landscape of the reduced Lennard–Jones potential.

son between SA and CGU searches for proteins with 5, 9, 16, 23, 30, and 36 residues (N residues correspond to $2N - 2$ degrees of freedom in this model). For short chains, both methods search to a nearly equal depth, but as chain length grows, SA gets increasingly stuck higher on the landscape relative to the CGU. Also, the CGU consistently finds solution that are similar in conformation, while the SA method generally does not. Random selection of points and subsequent minimization on an energy landscape with eight degrees of freedom without the use of the underestimator results in solution that are only 75% as deep as those found by the CGU, and gets worse with increasing dimensionality.

The protein landscape we studied may be a single funnel, while the van der Waals cluster landscape has several funnels. Comparison of Figures 2 and 4 highlights the greater advantage of the CGU method over the SA method for single vs. multifunnel landscapes as a function of the number of degrees of freedom. This increase in performance arises because the d_is are reasonably large, and the width of the bounding box is typically one-tenth that of the original for each degree of freedom. Thus, the CGU focusses only in regions relatively close to the global minimum. The final d_is, therefore, give a good indication of how funnel-like the landscape is.

Summary

We have compared the CGU underestimator search strategy to traditional trajectory-based search methods, MC, SA, and random selection with minimization, on two different types of energy landscapes. Our aim has been to compare methods for general search problems on funnel landscapes, so we have not used problem-specific optimization with any of the search methods. For landscapes having multiple equivalent funnels, the CGU is about equivalent to its underlying local minimum sampling strategy, which in this case is random selection with local minimization. But we find that the CGU is more efficient and more consistent in reaching lower energies on funnel-like landscapes than the other methods. This efficiency increases with the size of the problem. The three principal advantages of the CGU are that: (1) it uses all the information from the whole search, not just from the previous few samples, (2) it focusses and localizes the search by strongly restricting the bounds of its search. The parameters returned at the end of the search indicate the degree to which the landscape is funnel-shaped. (3) There are few problem-specific search parameters to be tuned, and therefore, the canonical CGU method may be generally applicable, with little or no optimization required. In addition, the CGU consistently finds its solutions in polynomial time, independent of any time-saving tricks. Thus, the CGU may be generally useful for global optimization searches in computational chemistry and biology.

Acknowledgments

K.W.F. thanks Jane Chang for helpful discussions. We thank Robert Leary for providing data on Lennard–Jones clusters using the Big Bang method and Jolanda Schreurs for help in making the figures.

References

1. Metropolis, N.; Rosenbluth, A. W.; Rosenbluth, M. N.; Teller, A. H.; Teller, E. J. J Chem Phys 1953, 21, 1087.
2. Ingber, L.; Mathemat Comp Modelling 1993, 18, 29.
3. Maranas, C. D.; Floudas, C. A. J Chem Phys 1994, 100, 1247.
4. Androulakis, I. P.; Maranas, C. D. Floudas, C. A. J Glob Opt 1997, 11, 1.
5. Maranas, C. D.; Androulakis, I. P.; Floudas, C. A. DIMACS Series in Discrete Mathemat Theoret Comput Sci 1996, 23, 133.
6. Phillips, A. T.; Rosen, J. B.; Walke, V. H. DIMACS Series in Discrete Mathemat Theoret Comput Sci 1996, 23, 181.
7. Dill, K. A.; Phillips, A. T.; Rosen, J. B. Dev Glob Opt 1997, 217.
8. Dill, K. A.; Phillips, A. T.; Rosen, J. B. J Comp Biol 1997, 4, 227.
9. Dill, K. A.; Chan, H. S. Nat Struct Biol 1997, 4, 10.
10. Doye, J. P. K.; Wales, D. J. J Chem Phys 1996, 105, 8428.
11. Ma, J.; Straub, J. E. J Chem Phys 1994, 101, 533.
12. Wille, L. T. Chem Phys Lett 1987, 133, 405.
13. Li, Z.; Scheraga, H. A. Proc Natl Acad Sci USA 1987, 84, 6611.
14. Sun, S.; Thomas, P. J.; Dill, K. A. Protein Eng 1995, 8, 769.